Eriantys Protocol Documentation

May 3, 2022

1. **Login Message**

This message is sent from the client to the server after establishing a connection to create a player session.

**Arguments**

* nickname: the name of the user

**Possible Responses**

* Accept Message: when the login is successful
* Error Message: this message specifies the type of error

1. **Ping** **Message**

This message is used from client and server to perform ping function.

**Arguments**

This message has no arguments.

**Response**

Ping Message.

1. **Accept Message**

This message is sent from the server to the client as a generic accept request message.

**Arguments**

This message has no arguments.

**Response**

This message has no responses.

1. **Enter Game Message**

This Message is sent from the client to the server when a player wants to start a new game.

**Arguments**

* Game type
* Number of Player

**Possible Responses**

* Accept Message
* Error Message

1. **Update Message**

This message is sent from the server to the client when the player needs the entire model (for example at the beginning of the game).

**Arguments**

* Model

**Response**

This message has no responses.

1. **Error Message**

This message is sent from the server to the client when a generic error occurred, specifying the error type. It also includes the specific string that the client has to show to the player.

**Arguments**

* Message Error String
* Error Type

**Response**

This message has no responses.

1. **End Game Message**

This message is sent from the server to the client when the game ends.

**Arguments**

* List of winner team’s color.

**Response**

This message has no responses.

1. **Character Message**

This message is sent from the client when it wants to play a character and from the server when a character has been played and clients need to modify their model. “Character Message” is an abstract message class that is extended by the specific character card’s class with its specific arguments.  
Every specific character’s class overrides the method “toArray” that return an array of Object, creating the specific array with its arguments.  
For example: Mushroom’s “toArray” has the color to eliminate from dominance strategy, Minstrel’s “toArray” has two student buffers containing the students from the dining room to swap with the students from entrance.

There are 12 specific character message implementations, one for every character: HERALD, KNIGHT, CENTAUR, MUSHROOM, JESTER, THIEF, MINSTREL, MONK, GRANNY\_HERBS, POSTMAN, PRINCESS, FARMER.

**Arguments**

* Card’s ID
* CharacterEnum’s value

**Response**

* Character Message
* End Game Message
* Error Message

1. **Play Assistant**

This message is sent from the client to the server when the player wants to play a new assistant card.

**Arguments**

* AssistantID: the identifier of the assistant card

**Response**

* Play Assistant: this message will modify client’s model
* Error Message: this message specifies the type of error

1. **Move Student**

This message is sent from the client to the server when the player wants to move some students from the entrance to an island or into the dining room.

**Arguments**

* toDing: the list of students to move into the dining room
* toIslands: the list of students to move on an island

**Response**

* Move students: this message will modify client’s model
* Error Message: this message specifies the type of error

1. **Choose Cloud**

This message is sent from the client to the server when the player wants to choose a cloud from which take the students.

**Arguments**

* cloudID: the identifier of the cloud

**Response**

* Choose Cloud: this message will modify client’s model
* Error Message: this message specifies the type of error
* End Game Message: this message contains the winner

1. **Move Mother**

This message is sent from the client to the server when the player wants to move mother nature from an island to another one.

**Arguments**

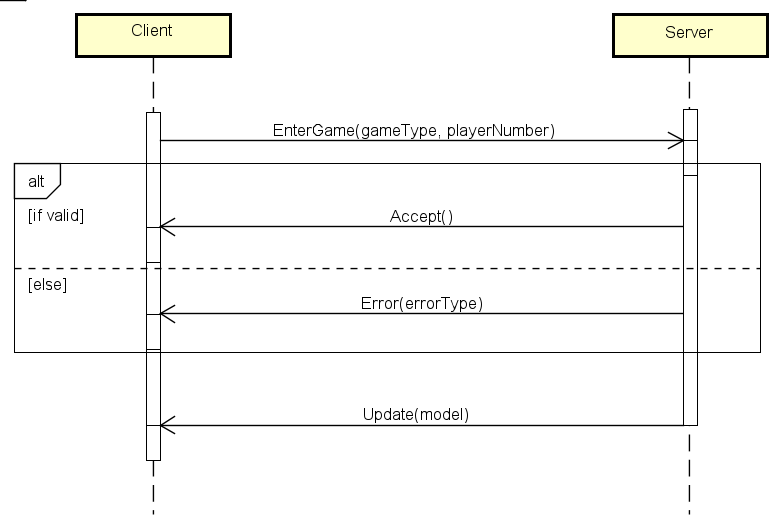
* step: number of steps that mother nature has to take between the islands

**Response**

* Move Mother Message: this message will modify client’s model
* End Game Message: this message contains the winner
* Error Message: this message specifies the type of error

Immagine che contiene tavolo

Descrizione generata automaticamenteLogin

Access

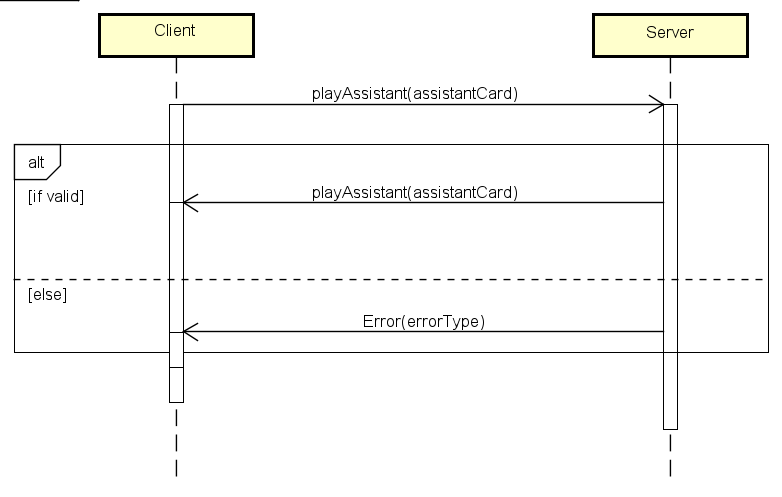
Planning

Immagine che contiene tavolo

Descrizione generata automaticamenteAction 1

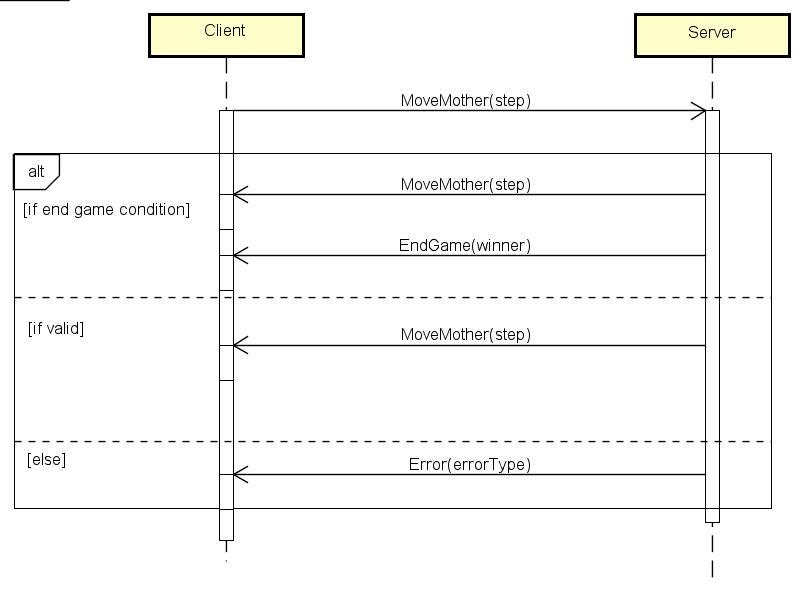
Action 2

Immagine che contiene tavolo

Descrizione generata automaticamenteAction 3